5. 5,423,025, Jun. 6, 1995, Error handling mechanism for a controller having a plurality of servers; Gary S. Goldman, et al., 714/57 [IMAGE AVAILABLE]

US PAT NO: 5,423,025 [IMAGE AVAILABLE] L3: 5 of 8
DATE FILED: Sep. 29, 1992

ABSTRACT:
An error handling and reporting mechanism is capable of taking advantage of sophisticated error analysis performed after clocks have been stopped in response to an error detected in a controller. The controller provides services in a data processing system in response to requests for controller services from a plurality of requestors. The controller includes a plurality of ports for storing requests for controller services. A plurality of servers is coupled to the plurality of ports, and perform separate services associated with the requests for controller services stored in the plurality of ports. An error reporting mechanism is included which is responsive to a detected error in a particular server associated with a request in a particular port, for posting error status in the particular port and causing clock stoppage within a clock stop latency period. An error analysis mechanism analyzes the detected errors during the clock stoppage. Error handling logic is coupled with the error analysis mechanism, and is responsive to the posted error status in the ports, for notifying a requestor of an error status posted with a request in the particular port. The error handling logic includes a stall counter, which stalls the error handling mechanism in response to the posted error status for at least the clock latency period so that the clock stoppage occurs and the error analysis mechanism completes error analysis before the requestor is notified.
US-CL-CURRENT: 714/57

ABSTRACT:
An error handling and reporting mechanism is capable of taking advantage of sophisticated error analysis performed after clocks have been stopped in response to an error detected in a controller. The controller provides services in a data processing system in response to requests for controller services from a plurality of requestors. The controller includes a plurality of ports for storing requests for controller services. A plurality of servers is coupled to the plurality of ports, and perform separate services associated with the requests for controller services stored in the plurality of ports. An error reporting mechanism is included which is responsive to a detected error in a particular server associated with a request in a particular port, for posting error status in the particular port and causing clock stoppage within a clock stop latency period. An error analysis mechanism analyzes the detected errors during the clock stoppage. Error handling logic is coupled with the error analysis mechanism, and is responsive to the posted error status in the ports, for notifying a requestor of an error status posted with a request in the particular port. The error handling logic includes a stall counter, which stalls the error handling mechanism in response to the posted error status for at least the clock latency period so that the clock stoppage occurs and the error analysis mechanism completes error analysis before the requestor is notified.

SUMMARY:

BSUM(11)

Accordingly, the present invention can be characterized as a controller providing services in a data processing system in response to requests for controller services from a plurality of requestors. The controller includes a plurality of ports for storing requests for controller services. A plurality of servers is coupled to the plurality of ports, and perform separate services associated with the requests for controller services stored in the plurality of ports. Error detection logic is coupled with the plurality of servers. An error reporting mechanism is included which is responsive to a detected error in a particular server, while the particular server is performing a service associated with a request in a particular port, for posting error status in the particular port and issuing a clock stop signal which results in clock stoppage within a clock stop latency period. An error analysis mechanism is coupled with the controller for analyzing the detected errors during the clock stoppage. Error handling logic is coupled with the error analysis mechanism, and is responsive to the posted error status in the ports, for notifying a requestor of an error status posted with a request in the particular port. The error handling logic includes a stall counter, which stalls the error handling mechanism in response to the posted error status for at least the clock stop latency period so that the clock stoppage occurs and the error analysis mechanism completes error analysis before the requestor is notified. During the clock stoppage, the error analysis mechanism may have an effect on the classification of the error which is reported with the error notification.

DETDESC:

DETD(7)

The system controller includes a plurality of servers, including server 104-1, through server N 104-N. Also included in the plurality of servers is a move in server 105. An error detection mechanism, such as parity checkers 106-1 through 106-n, and 107, is coupled with the plurality of servers. The error detection mechanism includes logic for reporting the error including a signal to set port error status in the port subject of the request, a set local hold signal which is coupled back to the respective server which suffered the error, and an error signal which is coupled to error bundling logic 108. The error bundling logic 108 includes latches for storing error history, and other error analysis logic as may be suited to a particular design. The error bundling logic 108 also generates a global hold signal on line 109 and a clocks off signal on line 110 which results in clock stoppage within a clock stop latency period. The clock stop latency in a large scale computer system may be from 10 to 20 cycles.

DETDESC:

DETD(23)

Thus, the error handling mechanism includes hardware to provide local and global hold states following detection of an error by a server. Also, the pinch and flush logic 114 for nominalizing the system controller prior to accepting any retry request is included. Port error status, or other status information is used to formulate an error response to the requestor coupled to the scan facility for updating by the service processor. The stall counter mechanism in the move in server ensures that the response is made only after the service processor has had an opportunity to affect the port error status. Software in the service processor executes during clock stoppage to analyze any port error status information and alter the default error response as needed. The analysis performed by the service processor is addressed to any and all ports in the system controller.

DETDESC:

DETD(33)

FIG. 3 also illustrates the server error detection and reporting logic 275 and the pinch-flush logic 250 which is coupled to the logic 275 across lines 252, and receives commands from the service processor across line 251. The pinch-flush logic 250 controls the interface 200 to flush the system controller in the event of a malfunctioning CPU as described above with respect to FIG. 2.

DETDESC:

DETD(36)

The error handling server starts the stall counter in three cases. First, the stall counter is started when any error is detected in the system controller in response to the global hold signal. The server may or may not be operating on the damaged port at the time of the error, or more than one port may be damaged. By stalling the error handling server as soon as any error is detected, the code in the service processor has a better chance at repairing and isolating the damage.

CLAIMS:

CLMS(1)

What is claimed is:

1. A controller providing services in a data processing system in response to requests for controller services from a plurality of requestors, comprising:
a plurality of ports for storing requests for controller services;
a plurality of servers, coupled to the plurality of ports, performing services associated with the requests for controller services stored in the plurality of ports;
error detecting means, coupled with the plurality of servers, for detecting errors in respective servers;
error reporting means, coupled to the error detecting means and responsive to a detected error in a particular server in the plurality of servers, while the particular server is performing a service associated with a request in a particular port in the plurality of ports, for posting error status in the particular port and issuing a clock stop signal which results in clock stoppage within a clock stop latency period;
error analysis data means, coupled with the plurality of servers, for providing error data for analysis after clock stoppage; and
error handling means, coupled with the error analysis data means and with the plurality of ports and responsive to posted error status, for notifying a requestor of an error status posted with a request in the particular port, said error handling means including a stall counter for stalling notification to a requestor in response to the posted error status until clock stoppage occurs and error analysis of the error data supplied by the error analysis data means has been completed.

CLMS(3)

3. The controller of claim 2, wherein the stall counter is further responsive to detection of an error in the one server coupled with the error handling means.

CLAIMS:

CLMS(8)

8. A system controller performing data transfer services in a data processing system in response to requests from a plurality of requestors, comprising:
a plurality of ports for storing requests for data transfer services;
a plurality of servers, coupled to the plurality of ports, performing data transfer services associated with the requests stored in the plurality of ports;
error detecting means, coupled with the plurality of servers, for detecting errors in respective servers;
error reporting means, coupled to the error detecting means and responsive to a detected error in a particular server in the plurality of servers performing a service associated with a request in a particular port in the plurality of ports, for posting error status in the particular port and issuing a clock stop signal which results in clock stoppage within a clock stop latency period;
a scan facility, coupled to the plurality of servers, for providing controller state information to a service processor during the clock stoppage for performing error analysis by the service processor; and
error handling means, coupled with the scan facility and with the plurality of ports and responsive to posted error status, for notifying a requestor of an error status posted with a request in the particular port, said error handling means including a stall counter for stalling notification to a requestor in response to the posted error status until clock stoppage occurs and error analysis is completed by the service processor.

CLAIMS:

CLMS(14)

14. A system controller performing data transfer services in a data processing system in response to requests from a plurality of requestors, comprising:
a request queue coupled to the plurality of requestors, including a plurality of ports for storing requests for data transfer services;
a plurality of servers, coupled to the plurality of ports, performing data transfer services associated with the requests stored in the plurality of ports, the plurality of servers including logic for holding service of a current request in response to a hold signal;
error detecting means, coupled with the plurality of servers, for detecting errors in respective servers;
error reporting means, coupled to the error detecting means and responsive to a detected error in a particular server in the plurality of servers performing a service associated with a request in a particular port in the plurality of ports, for posting error status in the particular port, for issuing a local hold signal to the logic for holding in the particular server, for issuing a global hold signal to the logic for holding in other servers in the plurality of servers, and for issuing a clock stop signal which results in clock stoppage within a clock stop latency period;
a scan facility, coupled to the plurality of servers, for providing controller state information during the clock stoppage to a service processor for performing error analysis by the service processor; and
error handling means, coupled with the scan facility and with the plurality of ports and responsive to posted error status, for notifying a requestor of an error status posted with a request in the particular port, including a stall counter for stalling notification to a requestor in response to a first occurrence of either the posted error status, the local hold signal or the global hold signal until clock stoppage and error analysis is completed by the service processor.

3. 5,708,775, Jan. 13, 1998, Fault information notification system localized at each network server; Jun Nakamura, 714/48, 47, 57 [IMAGE AVAILABLE]

US PAT NO: 5,708,775 [IMAGE AVAILABLE] L3: 3 of 8
DATE FILED: Oct. 17, 1995

ABSTRACT:
In a fault information notification system for notifying a manager unit of faults detected by a server unit on a network system in which a plurality of server units and the manager unit which manages the server units are connected in a network, each server unit comprises a fault information producing unit for producing fault information for various faults detected by the server unit to which sequence numbers are assigned, a fault recording unit for recording respective information in an extractable data structure for each fault information, and a fault history search unit for searching corresponding fault history information from the fault recording unit in response to a fault history search request including the reference numbers from the manager unit.
US-CL-CURRENT: 714/48, 47, 57

ABSTRACT:
In a fault information notification system for notifying a manager unit of faults detected by a server unit on a network system in which a plurality of server units and the manager unit which manages the server units are connected in a network, each server unit comprises a fault information producing unit for producing fault information for various faults detected by the server unit to which sequence numbers are assigned, a fault recording unit for recording respective information in an extractable data structure for each fault information, and a fault history search unit for searching corresponding fault history information from the fault recording unit in response to a fault history search request including the reference numbers from the manager unit.

SUMMARY:

BSUM(8)

In order to solve the above objects, according to a first aspect of the invention, in a fault information notification system for notifying a manager unit of faults detected by a server unit on a network system in which a plurality of server units and the manager unit for managing the server units are connected to the network, the network fault information notification system is characterized in that each server unit comprises a fault information producing means for producing fault information to which sequence numbers are assigned, with respect to various faults detected by the server unit, a fault recording means for recording respective information in an extractable data structure for each fault information, and a fault history search means for searching corresponding fault history information from the fault recording means in response to a fault history search request including the reference numbers from the manager unit.

SUMMARY:

BSUM(13)

In the fault information notification system of the present invention provided with such features, in a network system which connects a plurality of server units and a manager unit for managing the server units in a network, information on faults detected by a server unit is reported to the manager unit. Here, fault information producing means, fault recording means and fault history search means are provided in these server units. The fault information producing means produces fault information with sequence numbers attached thereto with respect to various faults detected by the server unit, and the produced fault information is recorded in an extractable data format for each fault information by the fault recording means. Then, the fault history search means searches corresponding fault history information from the fault recording means and responds according to a fault history search request including the sequence number from the manager unit.

SUMMARY:

BSUM(14)

In this way, information on various faults detected by each server unit is each given with sequence numbers, recorded and managed. By such means, according to a fault history search request including a sequence number from the manager unit, information on the history of the fault can be easily obtained. Also, in the fault information notification system of the present invention, where the

fault history search request includes a plurality of sequence numbers, the fault history search means searches a plurality of fault history information according to the fault history search request and responds. For this reason, fault histories for various server units can be easily obtained from the manager unit.

SUMMARY:

BSUM(15)

Also, in the fault information notification system of the present invention, the server units are further provided with an destination registration means and notification means. By means of the destination registration means, according to a fault notification request from the manager unit, the manager unit is registered as the destination of fault notification, whereupon the notification means, after detecting the fault, reports the fault information produced with sequence number attached to the manager unit registered in the destination registration means. By these means, since where a fault is detected it is reported to the manager unit which is already registered, the necessity for the manager unit to continuously supervise the server units is eliminated. Where notification of a fault is not necessary, a notification cancellation request is transmitted to server unit for which the notification is unnecessary, whereupon the destination registration means cancels the registration of the registered manager unit according to the notification cancellation request from the manager unit.

DETDESC:

DETD(2)

Herebelow, preferred embodiments of the present invention will be concretely described with reference to the drawings. FIG. 1 is a diagram illustrating the system structure of the fault information notification system according to the first embodiment of the present invention. In FIG. 1, 10 is a network communication path such as a LAN (Local Area Network) or the like, 11 is a client unit, 12 is a manager unit, and 13 is a server unit. The fault information notification system here is constructed in a network system in which a plurality of server units 13 and a manager unit 12 for managing the server units are connected to the network communication path 10. For this reason, in addition to a service processing section for normal service, various system components as illustrated in FIG. 2 are provided in the server unit 13. Faults detected by the various server units 13 on the system are reported to the manager unit 12 which manages the server units 13. In a state of normal system operation, the client unit 11 performs a processing request directly to the server units 13 according to a request for respective processing contents, and in the case of a processing request for the plurality of server units 13, performs the processing request via the manager unit 12.

DETDESC:

DETD(4)

In FIG. 2, 11 is a client unit, 12 is a manager unit, 13 is a server unit, 21 is an destination receiving section 21, 22 is a search receiving section 22, 23 is a fault history search section, 24 is a fault detection section, 25 is a fault information producing section, 26 is a fault information recording section, 27 is an information notification section, 28 is a destination registration section, and 29 is a service processing section.

DETDESC:

DETD(6)

The service processing section 29 is a processing section for executing primary service processing provided by the server unit 13. The fault detection section 24 detects error information from each of the service processing sections 29 or other faults of the relevant server unit 13. The content of the fault detected here is passed to the fault information producing section 25. In the fault information producing section 25, fault information is produced according to the contents of the detected fault and this fault information is passed to the fault information recording section 26. In the fault information recording section 26, upon fault information being passed, the sequence number of the relevant fault information is determined, this is recorded in a sequence number column of the fault information, and the relevant fault information is recorded in a log file. Then the fault information is passed to the information notification section 27. In the information notification section 27, the passed fault information is reported to the manager unit 12 which is the destination recorded in the destination registration table of the destination registration section 28.

CLAIMS:

CLMS(1)

What is claimed is:

1. A fault information notification system, comprising:
a network communication path;

a plurality of **server** units connected to said network communication path;
a **manager** unit connected to said network communication path and managing said **server** units, **faults detected** by said **server** units being notified to said **manager**;
wherein each of said **server** units comprises:
fault information producing means for producing **fault** information for various **faults detected** by said **server** unit to which a sequence number is assigned;
fault recording means for recording the produced **fault** information in an extractable data structure; and
fault history search means for searching corresponding **fault** history information from said **fault** recording means in response to a **fault** history search request including said sequence number from said **manager** unit.

CLAIMS:

CLMS(4)

4. A **fault** information notification system comprising:
a network communication path;
a plurality of **server** units connected to said network communication path;
a **manager** unit connected to said network communication path and managing said **server** units, **faults detected** by said **server** units being notified to said **manager**;
wherein each of said **server** units comprises;
fault information producing means for producing **fault** information for various **faults detected** by said **server** unit to which a sequence number is assigned;
fault recording means for recording the produced **fault** information in an extractable data structure;
fault history search means for searching corresponding **fault** history information from said **fault** recording means in response to a **fault** history search request including said sequence number from said, **manager** unit;
destination registration means for registering said **manager** unit as a destination of **fault** notification according to a **fault** notification request from said **manager** unit; and
notification means for, in response to production of **fault** information by said **fault** information producing means, notifying said manger unit which has been registered in said destination registration means of the produced **fault** information to which the sequence number has been assigned.

CLAIMS:

CLMS(6)

6. A **fault** information notification method for notifying a **manager** unit of **faults detected** by a **server** unit on a network system in which a plurality of **server** units and the **manager** unit which manages the **server** units are connected in a network, wherein each **server** unit executes the steps of:
producing **fault** information for various **faults detected** by said **server** unit to which a sequence number is assigned;
recording the produced **fault** information in an extractable data structure; and
searching corresponding **fault** history information from said recorded information in response to a **fault** history search request including said sequence number from said **manager**.

7. 5,367,670, Nov. 22, 1994, Computer system manager for monitoring events and operating parameters and generating alerts; Ronald G. Ward, et al., **714/47**; 364/221.7, 241.2, 241.4, 264, 264.2, 265, 266.6, 285, DIG.1; 395/704 [IMAGE AVAILABLE]

US PAT NO: 5,367,670 [IMAGE AVAILABLE] L3: 7 of 8
DATE FILED: Feb. 4, 1994

ABSTRACT:
A system manager for a computer system. The system manager transparently monitors signals transferred between computer system components along a system bus and stores objects related to the monitored signals in an object space. Information related to operating conditions within the system can then be provided from the object space. Later, the object space can be updated and the updated object space used to provide updated information regarding the operating conditions of the system.
US-CL-CURRENT: **714/47**; 364/221.7, 241.2, 241.4, 264, 264.2, 265, 266.6, 285, DIG.1; 395/704

DETDESC:

DETD(10)

Addressing the specific signals being **monitored** by the system bus **manager** 22, the computer system bus 13 supplies certain signals to a bus **monitor** 44 which will help determine the state of the computer system board 13. These signals include interrupt request (or "IRQ") signals, data memory request (or "DRQ") signals and input/output (or "I/O") signals. In one embodiment of the invention, it is contemplated that the bus **monitor** 44 **monitors** the I/O signals although, in a further embodiment of the invention, it is contemplated that the bus **monitor** 44 **monitors** the supplied IRQ, DRQ and I/O signals. If the signals are active, then the corresponding system resources are being used. In this manner, these signals may be used to **monitor** the performance of the computer system board 13. Other signals supplied by the computer system bus 13, are utilized during object management to indicate alert conditions. For example, the absence of the refresh signal will generate an alert since the lack of refresh may cause the file **server** 12 to **fail**. Similarly, an indication of a memory parity **error** will cause the generation of an alert. Also innately **monitored** by the bus **monitor** 44 are the printer port, so that the system **manager** 22 can report whether or not there is a printer **error** or is out of paper, the asynchronous serial port, so that the system **manager** can **monitor** and log asynchronous activity such as overrun **errors**, parity **errors**, and framing **errors** for system beard serial ports, system software, so that software **errors** can be identified, and keyboard events, so that keystrokes can be logged and the relationship between a system **failure** and keyboard inputs can be analyzed. Finally, the bus **monitor** 44 will **detect** the assertion of IOCHK, indicative of a catastrophic board **failure**, and board "times out", indicative of a violation of EISA standards. The bus **monitor** 44 transfers these signals to information processing and alert determination elements 52 where the **monitored** information is processed. As will be more fully described below, the information processing and alert determination elements 52 of the system **manager** 22 is comprised of a control processor and supporting logic which, by the application of object management techniques, is configured to determine whether the **monitored** information warrants the generation of an alert.

DETDESC:

DETD(16)

In addition to alert determination and generation based upon the passively **monitored** information, the information processing and alert determination elements 52 also perform several other functions. More specifically, the received information is also time stamped and stored or "logged" into RAM memory for later access. Thus, in the event of a catastrophic **failure** of the file **server** 12, the **monitored** and logged information will be available for "post mortem" diagnostics. Similarly, network information may be transferred over the bus master interface 46 and logged into RAM memory contained within the information processing and alert determination elements 52. Finally, the objects can be transferred, for example to the remote system **manager** facility 34 or the local network **manager** console 36 to provide real-time information regarding the performance of the system **manager** 22.